

A METHOD AND APPARATUS FOR INTERPRETING INFORMATION

5

Background of the Invention

Field of the invention

This invention relates to a method and apparatus for interpreting 10 information and particularly for information relating to a communications network.

Description of the prior art

In the telecommunications field, large amounts of data are available, 15 for example about customer behaviour and telephone usage. This data contains potentially useful information for many purposes such as detection of fraud, marketing, billing, maintenance planning and fault detection. However, the data must first be analysed in order to extract features that can easily be used for a given task. This task of extracting 20 useful features from the data is often difficult because the user does not know which type of features to look for. For example, the information may be in the form of call detail records (CDRs). A CDR is a log of an individual telephone call which contains information such as the length of the telephone call, the customer account number, the type of call and many 25 other pieces of information. Over a given time period many CDRs will be recorded, each containing many different pieces of information. When faced with this mass of information it can be difficult to know what features to extract for a particular problem.

One possibility is to use a data classifier which searches for a set of classes and class descriptions that are most likely to explain a given data set. Several types of such data classifiers are known. For example,

5 Bayesian classifiers, neural network classifiers and rule based classifiers. For a given task, a classifier is typically trained on a series of examples for the particular task. After the classifier has been trained then new examples are presented to it for classification. The classifier can be trained either using a supervised method or an unsupervised method. In a supervised 10 method the training examples that are used are known examples. That is the user knows which classes these training examples should be classified into and this information is also provided to the classifier during the training phase. For unsupervised training, there is no information about the desired classes for the training examples.

15

One problem is that the output of classifiers is often difficult to interpret. This is especially the case when unsupervised training has been used. The classifier output specifies which of a certain number of classes each input has been placed into. The user is given no explanation of what 20 the classes mean in terms of the particular task or problem domain. Neither is the user provided with any information about why a particular input has been classified in the way that it has.

Previously, users have needed to carry out complex analyses of the 25 classifier in order to obtain these kinds of explanations. Known examples can be input to the classifier and the outputs compared with the expected outputs. However, in order to do this known examples must be available

and this is often not the case. Even when known examples can be obtained this is often a lengthy and expensive procedure.

A further problem is that because these kinds of explanations are not available the user's confidence in the system is reduced. This means that the user is less likely to run the system, thus reducing the value of such a system. Also, errors and mistakes are hard to detect. For example, if erroneous data is entered by mistake a resulting error in the output could easily go unchecked. Similarly, if the training examples were not representative of the example population for the particular task then errors would be produced that would be hard to find.

It is accordingly an object of the present invention to provide an apparatus and method for interpreting information relating to a communications network which overcomes or at least mitigates one or more of the problems noted above.

Summary of the invention

According to a first aspect of the present invention there is provided a method of processing data relating to a plurality of examples using a data classifier arranged to classify input data into one of a number of classes, and a rule inducer, comprising the steps of:

- (i) inputting a series of inputs to the data classifier so as to obtain a series of corresponding outputs;
- 25 (ii) inputting said series of outputs and at least some of said series of inputs to the rule inducer so as to obtain a series of rules which describe relationships between the series of inputs to the data classifier and the series of corresponding outputs from the data classifier. This provides the

advantage that the rules can be used to provide an explanation for the user about how the classification is performed. Also, the rules can be used together with other information about the problem domain or task to help the user determine a "meaning" for each of the classes. Advantageously, 5 the user's confidence in the system is increased and errors in the system can more easily be detected and corrected.

Preferably, the data classifier is unsupervised. The output of an unsupervised classification system is especially difficult to interpret. 10 Advantageously, the rules produced according to the invention can be used to help the user determine a "meaning" for the output of the unsupervised classifier.

Preferably, the method further comprises the step of transforming the 15 series of rules into a format such that the formatted rules can be used as a data classifier. This provides the advantage that a rule based classifier can easily be created without the need for the user to determine the rules directly from the data set or other data source.

20 Preferably the method further comprises the step of incorporating the rules into a case-based reasoning system. This provides the advantage that a case-base reasoning system can easily be created without the need for the user to determine the rules directly from the data set or other data source. Advantageously the case-based reasoning system is able to learn 25 from new examples.

According to a second aspect of the present invention there is provided a method of processing data relating to a communications

network using a rule extractor and a neural network data classifier comprising the steps of:

- (i) inputting a series of training data inputs to the neural network and training the neural network using this series of training data so as to obtain 5 a series of output values corresponding to the training data inputs;
- (ii) inputting information about the configuration of the trained neural network to the rule extractor so as to obtain a series of rules which describe relationships between the series of training data inputs and the series of output values. This provides the advantage that the rules can be 10 used to provide an explanation for the user about how the classification is performed. Also, the rules can be used together with other information about the problem domain or task to help the user determine a "meaning" for each of the classes. Advantageously, the user's confidence in the system is increased and errors in the system can more easily be detected 15 and corrected.

According to another aspect of the present invention there is provided a computer system for processing data relating to a communications network comprising:

- 20 a data classifier arranged to classify input data into one of a number of classes;
- a rule inducer;
- a first input arranged to accept a series of inputs to the data classifier;
- a first output arranged to provide a series of corresponding outputs 25 from the data classifier;
- a second input arranged to accept said series of outputs and at least some of said series of inputs to the rule generator; and

a second output arranged to output from the rule generator a set of rules which describe relationships between the series of inputs to the data classifier and the series of corresponding outputs from the data classifier. This provides the advantage that the rules can be used to provide an 5 explanation for the user about how the classification is performed. Also, the rules can be used together with other information about the problem domain or task to help the user determine a "meaning" for each of the classes. Advantageously, the user's confidence in the system is increased and errors in the system can more easily be detected and corrected.

10

According to another aspect of the present invention there is provided a computer system for processing data relating to a telecommunications network comprising:

- (i) a rule extractor;
- 15 (ii) a neural network data classifier;
- (iii) a first input arranged to accept a series of training data inputs to the neural network;
- (iv) a processor arranged to train the neural network using the series of training data inputs so as to produce a series of output values 20 corresponding to the training data inputs; and
- (v) a second input arranged to accept information about the configuration of the trained neural network to the rule extractor so as to produce a series of rules which describe relationships between the series of training data inputs and the series of output values. This provides the 25 advantage that the rules can be used to provide an explanation for the user about how the classification is performed. Also, the rules can be used together with other information about the problem domain or task to help the user determine a "meaning" for each of the classes. Advantageously,

the user's confidence in the system is increased and errors in the system can more easily be detected and corrected.

Brief description of the drawings

5 Figure 1 is a general schematic diagram of an arrangement for interpreting data.

Figure 2 is a general schematic diagram indicating how a rule inducer and a data classifier are positioned in the arrangement of figure 1.

10 Figure 3 is a general schematic diagram indicating how a rule extractor and a data classifier are positioned in the arrangement of figure 1 according to another embodiment of the invention.

Figure 4 shows the use of the invention for detecting and analysing telecommunications fraud.

Figure 5 shows example attributes.

15 Figure 6 shows an example of input data for the rule inducer.

Figure 7 shows an example of output from the rule inducer.

Figure 8 shows the output of figure 7 incorporated into a rule-based classifier.

20 **Detailed description of the invention**

Embodiments of the present invention are described below by way of example only. These examples represent the best ways of putting the invention into practice that are currently known to the Applicant although they are not the only ways in which this could be achieved.

25

Definitions

rule extractor - any mechanism or technique for generating a set of rules to describe the relationship between the inputs and outputs of a

trained neural network that use information about the weighted connections in the neural network.

rule inducer - any mechanism or technique for generating rules to describe a plurality of data that involves generalising from the data.

5 data classifier - any mechanism or technique for dividing or breaking up a collection of data into groups.

self organising map (SOM) - a neural network architecture which discovers patterns in data by clustering similar inputs together. The data is grouped by the SOM without any prior knowledge or assistance. Grouping
10 is achieved by mapping the data onto a 2-D plane.

Figure 1 shows a computer system 1 that is arranged to automatically determine a classification system for a given data set that is provided to the system. The computer system accepts input data from a
15 data source 2. The computer system searches for a set of classes 3 and class descriptions that are most likely to explain the provided data set. Once the classification system has been determined, new data can be input and classified according to this system. For example, in a situation in which information about telephone calls needs to be analysed to detect
20 fraud, the data source 2 consists of information about individual telephone calls made during a certain time period. The computer system 1 determines a classification system and classifies the calls into a number of classes 3. Once this is done, a human operator or user then analyses the classes to see whether fraudulent calls appear only in certain classes. The
25 user obtains an explanation of how that data from the data source 2 has been classified as well as an explanation of what the classes 3 mean in terms of the particular data source 2 and the task or problem (e.g. fraud detection) involved.

In order to provide these explanations, figure 1 shows how the computer system 1 is also arranged to produce rules 4 which describe relationships between the input data from the data source 2 and the 5 classes 3. Advantageously, these rules 4 can then be used to provide an explanation of how the computer system 1 classified the input data 2. For example, such an explanation could be, "telephone call number 10 is a member of class 2 because it has feature A and feature B but not feature C". The rules 4 can also be used together with other information to assign 10 a "meaning" to the classes 3 that the input data 2 is classified into. For example, classes could be assigned meanings such as "fraudulent examples" and "non-fraudulent examples".

In one example, as shown in figure 2, the computer system 1 15 comprises a data classifier 21 and a rule inducer 25. A series of input data from a data source 22 is input to the data classifier 21 to produce a corresponding set of outputs 23. These outputs comprise information about which of a number of classes 23 each input is a member of. The series of input data from the data source 22 is also input to the rule inducer 20 25 as indicated by arrow 26. The rule inducer 25 also receives information about the corresponding series of outputs from the data classifier 21 as indicated by arrow 27. Given these inputs 26, 27 the rule inducer 25 produces a series of rules 24 which describe relationships between the series of input data provided to the data classifier 21 and the 25 corresponding series of outputs produced by the data classifier.

In an alternative example a rule extractor is used instead of a rule inducer. This is illustrated in figure 3. In this case the computer system 1

comprises a neural network data classifier 31 and a rule extractor 35. A series of training data 32 is input to the neural network 31 and the neural network is trained using this input data. The neural network 31 produces a series of outputs 33 or classes which correspond to the series of training 5 data. A description of the trained neural network 36 is provided to the rule extractor 35 which is then able to produce a series of rules 34. A description of the inputs 32 to the data classifier 31 may also be required as input to the rule extractor 35. These rules 34 describe relationships between the series of training data 32 and the corresponding series of 10 outputs 33 from the neural network. Any type of rule extractor can be used.

Once the rules 34, 24, 4 have been obtained they can also be used to create a rule-based classifier. This can then be used instead of or as well as the data classifier 21, 31. The rules 34, 24, 4 can also be 15 incorporated into a case-based reasoning system. A case-based reasoning system is advantageous in that it is able to learn by analogy.

A rule inducer is a fundamental component for a case-based reasoning system. Once the computer system 1 has been set up for a 20 particular application, such as for telecommunications data, then the system 1 can be incorporated into a case-based reasoning system. This enables a case-based reasoning system that is suitable for the particular application concerned to be set up quickly and easily.

25 The computer system 1 can be used to analyse data about the transmission of messages in a communications network. For example, the use of the computer system 1 to interpret data about the performance of EDNA is now described.

EDNA is a system which organises and effects the distribution of messages in a network of computers. It is a UNIX based distribution framework and it can be used to distribute software and files. It can also be
5 used for system management, file collection and process automation.

In this example, the task or aim is to investigate whether users of EDNA fall into distinct groups for billing purposes. If any groups are found it is also desired to explain characteristics of the groups and relate the
10 groups to the problem domain.

In this example, the data source 2 comprises information about the use of EDNA over a certain time period. For each user of EDNA a list of attribute values is given. A user of EDNA can be defined in different ways.
15 For example, a user could correspond to a department in a work place. It could comprise a number of different human users and/or nodes in the network. The list of attribute values for a user comprises information such as the number of files transferred by the user during the time period. In this example, 6 attributes are used as shown in figure 5. These include:
20 the number of EDNA transfers made during the time period 51;
the number of packages attempted during the time period 52;
the number of packages completed during the time period 53;
the number of links made during the time period 54;
the number of files transferred during the time period 55;
25 the total number of bytes transferred during the time period 56.

These attributes all relate to past usage of EDNA and comprise figures indicating what EDNA has done over a certain time period.

This data about the use of EDNA is then classified using a data classifier 21 that automatically searches for a set of classes 3 and class descriptions that are most likely to explain the data. Several different 5 classifiers can be used for this. In this example the known classifier AUTOCLASS is used.

AUTOCLASS is an unsupervised classification system based on Bayesian theory. It has been developed by P. Cheesman and his 10 colleagues and is described in the following documents which are intended to be incorporated herein by reference:

- P. Cheesman, J. Stutz, "Bayesian Classification (AutoClass) Theory and Results," in "Advances in Knowledge Discovery and Data mining", U.M. Fayyad et al. Eds. The AAAI Press, Menlo Park, 1995.
- R. Hanson, J. Stutz, P. Cheesman, "Bayesian Classification Theory", Technical Report FIA-90-12-7-01, NASA Ames Research Centre, Artificial Intelligence Branch, May 1991.
- P. Cheesman, J. Kelly, M. Self, J. Stutz, W. Taylor, D. Freeman, "AutoClass: a Bayesian Classification system." In 20 proceedings of the Fifth International Conference on Machine Learning, 1988.
- P. Cheesman, M. Self, J. Kelly, J. Stutz, W. Taylor, D. Freeman, "Bayesian Classification." In seventh National conference on Artificial Intelligence, pages 607-611, Saint Paul, Minnesota, 1988.

25

AutoClass has been implemented in the C programming language and is publicly available on the internet together with the following basic

and supporting documentation which is also incorporated herein by reference:

- preparation-c.text
- search-c.text
- 5 • reports-c.text
- interpretation-c.text
- checkpoint-c.text
- prediction-c.text
- classes-c.text
- 10 • models-c.text

These documents are publicly available on the Internet. They are typically distributed together with the source code for AUTOCLASS by file transfer from the Internet.

15 In the example of the present invention being discussed, the data classifier 21 is AutoClass. Data from the data source 22 is first prepared for use by AutoClass as described in the document preparation-c. text, referred to above. In this example the data source 22 comprises a list of 6 20 attribute values for each user of EDNA over a certain time period. This data is processed or formatted to meet the requirements specified in preparation-c. text. This involves creating a number of files containing the data and other parameters specified by the user.

25 The AutoClass is then used to classify the data as described in search-c. text. The output of the data classifier 21, in this case AutoClass, comprises:

- (i) a set of classes 3 each of which is described by a set of class parameters, which specify how the class is distributed along the various attributes;
- 5 (ii) a set of class weights describing what percentage of EDNA users are likely to be in each class;
- (iii) for each EDNA user, the relative probability that it is a member of each class.

10 AutoClass repeatedly classifies the data to obtain several sets of such results which are then compared by the operator to determine the most successful classification(s).

15 When AutoClass is used various options can be chosen by the operator. These include parameters such as: specifying how many classes to look for or try; or specifying a maximum duration for the classification. Default values for these parameters are used unless the operator specifies otherwise. In the particular example being discussed, about EDNA, the parameter values were set so as to "look for" 20 classes.

20 However, it is not essential to use these exact parameter values. In different situations different values may be more appropriate. The various parameters are discussed in search-c. text as well as the other documents referred to above.

25 The several sets of results are ranked by AUTOCLASS and one set is chosen. Typically the classification which describes the data most completely is used. For example, this can be done by comparing the log total posterior probability value for each classification, as described in

interpretation-c.text. The results of the chosen classification can then be analysed using AutoClass by generating simple reports which display the results in different ways. This is described in reports-c.text. One of the reports that is generated contains the ranked sets of results.

5

As mentioned earlier AutoClass is only one possible system that can be used for the data classifier 21. Any classification technique that determines a set of classes for a given data set can be used.

10 In the AutoClass system, class membership is expressed probabilistically rather than as logical assignment. This is done by defining the classes in terms of parameterised probability distributions. Each example is considered to have a probability that it belongs to each of the classes. In this way, examples can be members of more than one 15 class. However, each example must belong to at least one class because AutoClass makes the class probabilities sum to 1.

Alternatively, clustering techniques could be used for the data classifier 21. Clustering techniques act to partition the data into classes so 20 that each example is assigned to a class. For example, one of the basic known approaches is to form a least squares fit of the data points to a pre-specified number of groupings. This requires that the number of clusters is known in advance, which is often not the case. Adaptive clustering techniques can also be used. These do not rely on predefined 25 parameters, for example see EP-A-0436913. Clustering techniques differ from the AutoClass and other Bayesian systems which search in a model space for the "best" class descriptions. A best classification optimally trades off predictive accuracy against the complexity of the classes and

does not "over fit" the data. Also, in AutoClass the classes are "fuzzy" so that an example can be a member of more than one class with different probabilities for membership.

5 Another possibility is to use a neural network classifier. This could either be unsupervised or supervised. Also, the neural network can be configured to give Bayesian probability outputs if desired using known techniques. Neural networks have a number of advantages that makes them particularly suitable for use with information about the transmission of
10 messages in a communications network and specifically for telecommunications applications. These advantages include:

- neural networks can be used to discover complex underlying patterns and anomalies in communications network data;
- neural networks can learn both the normal and fraudulent
15 communications behaviour from examples;
- neural networks can adapt to changes in the communications network data;
- neural networks can perform data analysis on many different variables;
- neural networks are excellent at performing pattern recognition tasks, such as detecting known behaviour types;
- neural networks are more resilient than standard statistical techniques to noisy training data;
- neural network technology only requires to be retrained
20 periodically;
- a trained neural network is able to process new communications data quickly.

Once a classification has been obtained, information about this classification 27 is combined with information about the data source 22 to provide input for the rule inducer 25. For example, in the EDNA example being discussed Figure 6 shows the combined data ready for input to the 5 rule inducer 25. Figure 6 shows a table of values 61 where each row in the table is for an EDNA user. The first 6 columns of the table 61, show the attribute values as input to the data classifier 21. The last column 61 shows the class that each EDNA user has been classified into using the data classifier 21.

10

Alternatively the output of the data classifier 21 may already be in a form similar to that shown in Figure 6. That is, information from the data source 22 may not need to be combined with the classification results 27.

15

In the example being discussed, about EDNA, the system used for the rule inducer 25 is CN2. This is a publicly available algorithm which is described in the following documents which are intended to be incorporated herein by reference:

- P. Clark and T. Niblett "The CN2 Induction Algorithm", in 20 Machine Learning Journal, 3 (4), pp 261 - 283, Netherlands: Cluner (1989)
- P. Clark and R. Boswell. "Rule Induction with CN2: Some Recent Improvements," In Machine Learning - Proceedings of the Fifth European Conference (EWSL - 91), pp 151 - 163, Ed: Y Kodratoff, 25 Berlin: Springer Verlag (1991).
- R. Boswell "Manual for CN2 version 6.1" (1990) The Turing Institute Limited IT/P2154/RAB/4/1-5

It is not essential to use the CN2 algorithm for the rule inducer 25; alternative rule induction techniques can be used. A rule inducer is a means by which a rule-based system can learn by example. The process of rule induction involves the creation of rules from a set of examples. The 5 idea is to create rules which describe general concepts of the example set. The term rule inducer is used here to refer to any system which involves the creation of rules from a set of examples.

CN2 is a rule induction algorithm which takes a set of examples (that 10 are vectors of attribute values and information about which class each example is a member of) and generates a set of rules for classifying them. For example, a rule might take the form: "If telephone call number 10 has attribute A and attribute B but not attribute C then it is a member of Class 2." 15

In the example being discussed about EDNA, the rules take the form shown in Figure 7. This shows 6 IF-THEN rules 71 and a default condition 72. The attribute names 73 correspond to the attributes shown in Figure 5 and the rules specify various threshold valves for the attributes 74. Each 20 rule has a THEN portion specifying a membership of a particular Class 75. The numbers in square brackets 76 indicate how many examples met the conditions of the rule and were assigned to the particular class.

The rule inducer 25 may either be of a kind which produces an 25 ordered sequence of rules or of a kind which produces an unordered sequence of rules. When an ordered rule sequence is produced, then when the induced rules are used to process new examples each rule is tried in order until one is found whose conditions are satisfied. Order

independent rules require some additional mechanism to be provided to resolve any rule conflicts which may occur. This has the disadvantage that a strict logical interpretation of the rules is detracted from. However, ordered rules also sacrifice a degree of comprehensibility as the 5 interpretation of a single rule is dependent on which other rules preceded it in the list. The CN2 algorithm can be configured to produce either unordered or ordered rules. In the particular example being discussed about EDNA, an unordered rule list was created.

10 The rules obtained from the rule inducer 25 are then evaluated. This is done by comparing the information that was provided as input to the rule inducer with the rules. For example in the EDNA situation input to the rule inducer 25 took the form as shown in Figure 6. For the first row in Figure 6 the attribute value for sumofsumofnum pack is 27 and for 15 sumofsumoffcomplete is 14 which satisfies the condition for the first rule in Figure 7. This example is assigned to Class C0 following the induced rules, and from Figure 6 we can see that it was also assigned to Class C0 by the data classifier. This type of evaluation can be carried out automatically using the CN2 system as described in the CN2 Manual 20 referred to above.

When a successfully evaluated set of rules is obtained this can be used to create a rule based classifier. For example, Figure 8 shows a program written in the programming language PERL. This program 25 incorporates the rules from Figure 7 that were produced by the rule inducer in the EDNA example. For example 81 shows one such rule. When this program is executed new examples are classified into one of the classes C0, C1, C2 or C3. In this way the program can be used to classify new

examples of EDNA use into one of these predetermined 4 classes. It is not essential that the programming language PERL is used; any suitable means for executing the rules can be used.

5 The rule based classifier shown in Figure 8 is static and does not "learn by example". That is, once the rules are induced and formed into the classifier they are not altered automatically. However, it is possible to incorporate the set of successfully evaluated rules into a more sophisticated rule based classifier. For example, this could be arranged to
10 learn from new examples that are presented to the classifier for classification. Also the rules or computer system 1 can be incorporated into a case based reasoning system. In this type of system learning by analogy plays an important role. Existing knowledge is applied to a new problem instance on the basis of similarities between them. This can
15 involve modifications of the existing knowledge to fit the new case. A case based reasoning system typically has a case base which comprises a set of relevant examples. These cases are applied to new problems by an analogical reasoning process.

20 A second example of the use of the invention is in detecting telecommunications fraud. As shown in Figure 4 call detail record data 41 is input to an anomaly detector 42 which produces information about which of the call detail records are fraud candidates 43. The anomaly detector 42 comprises several components including a kernel 44 which incorporates a
25 neural network. This neural network is trained to classify the input information 41 into classes indicating fraud or non-fraud candidates 43. The neural network is thus equivalent to the data classifier 21. A rule inducer 25, 45 is incorporated into the anomaly detector 42. The rule

inducer 45 receives output information from the neural network which comprises a set of attributes for each customer account together with a class assignment for that customer account. The rule inducer then generates rules 24, 46.

5

The neural network can be of many possible types. For example, a self-organising map or a multi-layer perceptron. For example, if a self organising map is used and the task is to detect telecommunications fraud many different classes may be produced by the neural network classifier.

10 Some of these classes may relate to known types of fraud and others to legitimate use. Still further classes may relate to unknown examples which could be new types of fraud or new types of legitimate use. When a new type of fraud evolves it is important for the operator to react to this quickly. A new "unknown" class may emerge in the self organising map which

15 could contain new types of fraud. By using the output of the rule inducer, or extractor, the operator can quickly obtain information about the characteristics of the new class.

A wide range of applications are within the scope of the invention.

20 For example, interpreting information relating to telecommunications fraud; credit card fraud; faults in a communications network and encryption key management. The invention applies to any situation in which a large amount of data needs to be analysed to extract features necessary for a particular task or problem domain and where it is required to explain or

25 interpret the way in which the features were obtained. This can be used for knowledge engineering in the development of expert systems. The invention also applies to pattern recognition tasks for example taxonomy in biology, object recognition and object tracking.